

**This Page is Inserted by IFW Indexing and Scanning
Operations and is not part of the Official Record**

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.

Remarks

Claims 1-22 are pending in the application. All claims stand rejected. By this paper, claims 1 and 2 have been amended. Claims 10-22 have been cancelled. New claims 23-40 have been added to provide claim coverage commensurate with the scope of the invention.

Claims 1-6, 9-11, 14-19, and 21-22 were rejected under 35 U.S.C. 102(e) as being anticipated by Voyticky et al. ("Voyticky"). Claims 7-8, 12-13, and 20 were rejected as being unpatentable over Voyticky.

Claim 1 has been amended to recite a method comprising:

presenting a broadcast segment as part of an interactive television transmission;

receiving with the broadcast segment supplemental information related to a transaction involving the broadcast segment;

responsive to a first command received from a user input device, locally storing the supplemental information received prior to the first command; and

responsive to a second command received from the user input device and subsequent to presenting at least a portion of the broadcast segment, retrieving the locally-stored information associated with the transaction and using the retrieved information to resume the transaction.

These claimed features allow a broadcaster to include supplemental information, such as ATVEF triggers, within a broadcast segment to enable various electronic transactions and enhanced content. In response to a first user command (e.g., to defer the transaction), the system locally stores the supplemental information that was received prior to the command. At a later time, the locally-stored information can be retrieved to resume the transaction.

By contrast, Voyticky does not receive supplemental information, such as an ATVEF trigger, with a broadcast segment; nor does Voyticky locally store supplemental information received prior to a user command to defer the transaction. Instead, Voyticky merely sends a time index to a central server, the time index representing the time at which an "event" button was pressed on the remote control. Thereafter, "the server determines an assortment of goods and services that were displayed on the user's television" when the button was pressed (see Abstract).

In this regard, the claimed invention is radically different from Voyticky's approach. For example, the claimed invention does not require a central server to monitor potentially thousands or millions of button presses on individual remote controls. Indeed, the claimed invention does not even require a central server. For instance, using the claimed methodology, a set top box could receive a trigger including a URL directed to a manufacturer's website. The set top box locally stores URL so that it can be retrieved locally, at any time, without having to rely on an intermediary as in Voyticky. If Voyticky's central server is down, a user cannot initiate a transaction, even if the manufacturer's website is available.

There is no teaching or suggestion in Voyticky of receiving supplemental information, such as ATVEF triggers, with a broadcast segment to enable a transaction. Indeed, Voyticky is directed in completely the opposite direction, avoiding embedded triggers in favor of using time indices and a database within a central server. Thus, Voyticky actually teaches away from the claimed invention.

In view of the foregoing, the applicant respectfully submit that claim 1, as amended, is patentably distinct over the cited reference. Claims 2-9 depend directly

or indirectly from claim 1 and are likewise believed to be patentably distinct for at least the same reasons.

New claim 23 recites a method, comprising:

receiving a broadcast segment including supplemental information sufficient for conducting at least a portion of a transaction;

notifying a user that the transaction is available;

receiving a user command to defer the transaction;

storing the supplemental information;

storing context information relating to the transaction;

deferring the transaction;

receiving a user command to resume the deferred transaction;

retrieving the stored supplemental information and context information; using the supplemental information and context information to resume the deferred transaction from the point at which it was deferred and restore the user's context within the transaction.

These claimed features allow a transaction to be deferred by storing supplemental information, such as triggers received with the broadcast, which can be subsequently retrieved to resume the transaction from the point at which it was deferred. In addition, the system stores context information relating to the transaction, such as any information previously entered by a user in connection with the transaction or snapshot of the broadcast segment relating to the transaction. The context information can be later retrieved when the transaction is to be resumed in order to restore the user's context within the transaction.

Voyticky does not disclose storing both supplemental information and context information. At best, Voyticky discloses storing a time index indicating when the

"event" button was pressed. This time index is not received with the broadcast segment as required by claim 23 and cannot, therefore, be the recited supplemental information. Voyticky does not disclose or suggest storing ATVEF triggers or URLs (as recited in new claims 24 and 25), or other mechanisms for enabling interactivity. Indeed, Voyticky appears to be presented as an alternative to such techniques.

New claim 26 recites the step of locally storing the supplemental information within a set top box. Voyticky does not locally store information, with the possible exception of a time index. The time index is not, by itself, sufficient to conduct at least a portion of a transaction, as required by new claim 23. A server must cross-reference the time index with schedule information in a database, after which the server provides information necessarily for conducting the transaction. Accordingly, Voyticky's time indices cannot be the claimed supplemental information.

New claim 27 recites the step of storing context information including information previously entered by a user in connection with the transaction. For example, the user may enter his name, address, etc., but not have his credit card available. The user may then defer the transaction, after which the system saves the previously-entered name, address, etc., as context information. When the user resumes the transaction, this context information may be retrieved so that the user does not need to reenter it.

By contrast, Voyticky does not actually defer transactions in the sense of interrupting a transaction that has been already initiated. Hence, there is never any previously-entered information in connection with the transaction, as required by claim 27. In Voyticky, a user presses an "event" button, which merely records the

time at which the button was pressed. Later, the user may decide to commence a transaction (assuming the central server has stored content for the time at which the user pressed the "event" button). The user never has the opportunity to enter any information prior to pressing the "event" button.

Similarly, new claims 28 and 29 recite storing context information including URLs and content of websites accessed in connection with the transaction. This allows the system to immediately resume the transaction without having to reacquire this information from an external source. By contrast, Voyticky teaches exactly the opposite, retrieving all of the content from an external source (the central server) at the time a user decides to begin a transaction that was previously marked by pressing the "event" button. As explained in connection with new claim 27, Voyticky does not interrupt transactions that are in progress. Accordingly at the time that the user presses the "event" button, there is no website content that has been retrieved or URLs that have been accessed.

New claim 30 recites the step of storing context information including at least one snapshot of the broadcast segment relating to the transaction. Voyticky does not disclose storing a snapshot of a broadcast after the "event" button is pressed. At best, Voyticky might store video clips or the like within the central server that can later be correlated with a time index.

Even if Voyticky could be construed as storing a snapshot in connection with the "event" button, it does not locally store the snapshot in a set top box, as required by new claim 31. At best, Voyticky stores video clips and the like within a central

server. Furthermore, Voyticky does not locally capture the a snapshot, as recited in new claim 32.

New claim 33 recites the step of storing context information including an indication of a current action within a set of actions of the transaction at which point the transaction is to be deferred. A transaction may comprise a process including a series of steps. The claimed invention allows the user to interrupt a transaction at any step in the process and later resume the transaction at the same step.

Unlike the claimed invention, Voyticky cannot interrupt a transaction in progress. Voyticky merely notes the time at which the "event" button is pressed and later correlates that time with a database. There is no transaction to interrupt when the "event" button is pressed. There is no teaching or suggestion in Voyticky of interrupting one of the individual steps within a transaction, as recited in claim 33.

New claim 35 recites the step of presenting an audio indicator of the availability of the transaction. Displaying a visual notification (e.g., icon) of an available transaction can be distracting and even a little annoying for a user, particularly where there may be a large number of available transactions. This claimed feature allows a user to be notified by an audio indicator, which can be made to be less distracting than a visual indicator. Voyticky does not disclose or suggest an audio indicator of the availability of a transaction.

New claim 36 recites the steps of displaying a list of deferred transactions and receiving a user selection of a deferred transaction to resume. In addition, new claim 37 recites displaying a list of previously-completed transactions and a list of cancelled

transactions. Voyticky does not disclose or suggest displaying a list of deferred transactions, previously-completed transactions, and cancelled transactions.

New claim 38 recites automatically displaying a list of deferred transactions during a commercial break, while new claim 39 recites automatically displaying a list of deferred transactions after the broadcast segment program being currently viewed has ended. These claimed features remove the need for a user to remember to view the list of deferred transactions. For example, the user may configure the system to automatically display the list each time a commercial break begins. No such teaching of automatically displaying a list of deferred transactions is provided or suggested by Voyticky.

New claim 40 recites a method, comprising:

- receiving a broadcast segment including supplemental information for conducting a transaction;

- receiving a command to defer the transaction;

- in response to the command to defer the transaction, capturing a snapshot of at least a portion of the broadcast segment relating to the transaction;

- locally storing the snapshot within a set top box;

- deferring the transaction;

- receiving a command to resume the deferred transaction;

- retrieving the locally-stored snapshot; and

- presenting the retrieved snapshot to restore a user's context in the transaction.

As explained above, Voyticky does not disclose capturing a snapshot of a broadcast segment in response to a command to defer a transaction. Furthermore,

Voyticky does not disclose locally storing the snapshot within a set top box. Finally, Voyticky does not disclose retrieving a locally-stored snapshot and presenting the retrieved snapshot to restore a user's context in the transaction.

In view of the preceding amendments and remarks, the applicant respectfully submits that claims 1-9, as amended, as well as new claims 23-40, are patentably distinct over the cited reference. A Notice of Allowance is respectfully requested.

Respectfully submitted,

Digeo, Inc.

By


Kory D. Christensen
Registration No. 43,548

STOEL RIVES LLP
One Utah Center Suite 1100
201 S Main Street
Salt Lake City, UT 84111-4904
Telephone: (801) 328-3131
Facsimile: (801) 578-6999